AMENDMENT

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1 – 29 (cancelled)

~~31~~30. (currently amended) A computer readable medium programmed to operate a method of removing an empty string term from an automaton A having a set of states "p" and a set of states "q", the method comprising:

computing an ε-closure for each state "p" of the automaton A;

modifying outgoing transitions of each state "p" by:

removing each transition labeled with an empty string; and

adding to each transition leaving "p" a non-empty-string transition, wherein each state "q" is left with its weights pre-⊗-multiplied by an ε-distance from state "p" to a state "q" in the automaton A.

~~32~~31. (currently amended) The computer readable medium of claim ~~31~~30, the method on the computer readable medium further comprising:

removing inaccessible states using a depth-first search of the automaton A.

~~33~~32. (currently amended) The computer readable medium of claim ~~31~~30, wherein the step of adding to E[p] non-empty-string transitions further comprises leaving q with weights (d[p,q] ⊗ w[e]) to the transitions leaving p.

~~34~~33. (currently amended) The computer readable medium of claim ~~31~~30, wherein the step of computing ε-closure for each input state of an input automaton A further comprises:

removing all transitions not labeled with an empty string from automaton A to produce an automaton A_ε;

decomposing A_ε into its strongly connected components; and

computing all-pairs shortest distances in each component visited in reverse topological order.

~~35~~34. (currently amended) A circuit programmed to operate a method of removing an empty string term from an automaton A having a set of states "p" and a set of states "q", the method comprising:

computing an ε-closure for each state "p" of the automaton A;

modifying outgoing transitions of each state "p" by:

removing each transition labeled with an empty string; and

adding to each transition leaving "p" a non-empty-string transition, wherein each state "q" is left with its weights pre-multiplied by an ε-distance from state "p" to a state "q" in the automaton A.

~~36~~35. (currently amended) The circuit of claim ~~35~~34, the method programmed into the circuit further comprising:

removing inaccessible states using a depth-first search of the automaton A.

~~37~~36. (currently amended) The circuit of claim ~~35~~34, wherein the step of adding to E[p] non-empty-string transitions further comprises leaving q with weights (d[p,q] ⊗ w[e]) to the transitions leaving p.

~~38~~37. (currently amended) The circuit of claim ~~35~~34, wherein the step of computing ε-closure for each input state of an input automaton A further comprises:

removing all transitions not labeled with an empty string from automaton A to produce an automaton A_ε;

decomposing A_ε into its strongly connected components; and

computing all-pairs shortest distances in each component visited in reverse topological order.

~~39~~38. (currently amended) A computer readable medium programmed to operate a method of removing an empty string term from a transducer A having a set of states "p" and a set of states "q", the method comprising:

computing an ε-closure for each state "p" of the transducer A;

modifying outgoing transitions of each state "p" by:

removing each transition labeled with an empty string; and

adding to each transition leaving "p" a non-empty-string transition, wherein each state "q" is left with its weights pre-⊗-multiplied by an ε-distance from state "p" to a state "q" in the transducer A.

~~40~~39. (currently amended) The computer readable medium of claim ~~39~~38, the method on the computer readable medium further comprising:

removing inaccessible states using a depth-first search of the transducer A.

~~41~~40. (currently amended) The computer readable medium of claim ~~39~~38, wherein the step of adding to E[p] non-empty-string transitions further comprises leaving q with weights (d[p,q] ⊗ w[e]) to the transitions leaving p.

~~42~~41. (currently amended) The computer readable medium of claim ~~39~~38, wherein the step of computing ε-closure for each input state of an input transducer A further comprises:

removing all transitions not labeled with an empty string from transducer A to produce a transducer A_ε;

decomposing A_ε into its strongly connected components; and

computing all-pairs shortest distances in each component visited in reverse topological order.

~~43~~42. (currently amended) A circuit programmed to operate a method of removing an empty string term from a transducer A having a set of states "p" and a set of states "q", the method comprising:

computing an ε-closure for each state "p" of the transducer A;

modifying outgoing transitions of each state "p" by:

removing each transition labeled with an empty string; and

adding to each transition leaving "p" a non-empty-string transition, wherein each state "q" is left with its weights pre-⊗-multiplied by an ε-distance from state "p" to a state "q" in the transducer A.

~~44~~43. (currently amended) The circuit of claim ~~43~~42, the method programmed into the circuit further comprising:

removing inaccessible states using a depth-first search of the transducer A.

~~45~~44. (currently amended) The circuit of claim ~~43~~42, wherein the step of adding to E[p] non-empty-string transitions further comprises leaving q with weights (d[p,q] ⊗ w[e]) to the transitions leaving p.

~~46~~45. (currently amended) The circuit of claim ~~43~~42, wherein the step of computing ε-closure for each input state of an input transducer A further comprises:

removing all transitions not labeled with an empty string from transducer A to produce a transducer A_ε;

decomposing A_ε into its strongly connected components; and

computing all-pairs shortest distances in each component visited in reverse topological order.

~~47~~46. (cancelled)

~~48~~47. (currently amended) The ~~automaton~~ computer-readable medium of claim ~~47~~54, the method of creating the automaton B further comprising:

removing inaccessible states using a depth-first search of the input automaton.

~~49~~48. (currently amended) The ~~automaton~~ computer-readable medium of claim ~~47~~54, wherein the step of adding to E[p] non-ε-transitions further comprises leaving q with weights (d[p,q] ⊗ w[e]) to the transitions leaving p.

~~50~~49. (currently amended) ~~A automaton~~ The computer-readable medium of claim ~~47~~54, wherein the step of computing an ε-closure for each input state of an input automaton A further comprises:

removing all transitions not labeled with an empty string from automaton A to produce an automaton A_ε;

decomposing A_ε into its strongly connected components; and

computing all-pairs shortest distances in each component visited in reverse topological order.

~~51~~50. (cancelled)

~~52~~51. (currently amended) The ~~automaton~~ computer-readable medium of claim ~~51~~55, the method of creating the transducer B further comprising:

removing inaccessible states using a depth-first search of the input transducer.

~~53~~52. (currently amended) The ~~automaton~~ computer-readable medium of claim ~~51~~55, wherein the step of adding to E[p] non-ε-transitions further comprises leaving q with weights (d[p,q] ⊗ w[e]) to the transitions leaving p.

~~54~~53. (currently amended) The computer-readable medium ~~A automaton~~ of claim ~~51~~55, wherein the step of computing an ε-closure for each input state of an input transducer A further comprises:

removing all transitions not labeled with an empty string from transducer A to produce a transducer A_ε;

decomposing A_ε into its strongly connected components; and

computing all-pairs shortest distances in each component visited in reverse topological order.

~~55~~54. (currently amended) A computer readable medium storing an executable automaton B having no ε-transitions, the automaton B produced according to a method of removing ε-transitions from an input automaton A having a set of states "p" and a set of states "q", the method comprising:

computing an ε-closure for each state "p" of the input automaton;

modifying outgoing transitions of each state "p" by:

removing each ε-transitions; and

adding to each transition leaving "p" a non-ε-transitions, wherein each state "q" is left with its weights pre-multiplied by an ε-distance from state "p" to a state "q" in the input automaton.

~~56~~55. (currently amended) A computer readable medium storing an executable transducer B having no ε-transitions, the transducer B produced according to a method of removing ε-transitions from an input transducer A having a set of states "p" and a set of states "q", the method comprising:

computing an ε-closure for each state "p" of the input automaton;

modifying outgoing transitions of each state "p" by:

removing each ε-transitions; and

adding to each transition leaving "p" a non-ε-transitions, wherein each state "q" is left with its weights pre-multiplied by an ε-distance from state "p" to a state "q" in the input transducer.